Beyond Adjacency Pairs: Hierarchical Clustering of Long Sequences for Human-Machine Dialogues

Maitreyee Tewari
Department of Computing Science
Umeå University
Umeå, Sweden
maittewa@cs.umu.se

Abstract

This work proposes a framework to predict sequences in dialogues, using turn based syntactic features and dialogue control functions. Syntactic features were extracted using dependency parsing, while dialogue control functions were manually labelled. These features were transformed using tf-idf and word embedding; feature selection was done using Principal Component Analysis (PCA). We ran experiments on six combinations of features to predict sequences with Hierarchical Agglomerative Clustering. An analysis of the clustering results indicate that using word-embeddings and syntactic features, significantly improved the results.

1 Introduction

Dialogues between humans is not a solitary activity of words, rather the involved participants have certain desires/goals that they want to achieve. In order to do that, they co-create understanding, by aligning aspects of their believes/knowledge to achieve their goals and reach a consensus using dialogue control functions. Dialogues between humans and machines can be facilitated by dialogue management systems (DMS). A basic DMS operates by coordinating natural language understanding (NLU), natural language generation (NLG) and a dialogue manager (DM). A DM employs either learned or hand-crafted strategies to the output from the NLU and sends its decisions to NLG that carries forward the interaction with the human participant.

A DM's flexibility can be partially attributed to the incoming knowledge from the NLU. By DM's flexibility we mean to have functions for anaphora resolution, co-referencing, keeping track of topic shifts and being able to return to previous topics (McTear et al., 2016).

The **motivation** behind this work is to explore sequences (Nicholas et al., 2016) in dialogues that can improve the NLU's knowledge. A well explored dialogue sequencing method (Palomar and Patricio, 2000; Boyer et al., 2009) with-in conversation analysis (CA) are studied as *adjacency pairs* (Schegloff and Sacks, 1973) such as *(Question-Answer, Request-Accept, Greeting-Greeting etc.)*, where the first one in the pair is called first pair part (FPP_{base}) and the second one is called second pair part (SPP_{base}). For exploring long sequences, CA provides a relevant framework of *sequence expansion* (Stivers, 2012) allowing the prior mentioned base parts to be expanded with preceding parts (FPP_{pre}, SPP_{pre}), insertion parts (FPP_{insert}, SPP_{insert}) or succeeding parts by (FPP_{post}, SPP_{post}).

This work proposes to use sequence expansion to analyse how much long sequences can be predicted by the machine learning models in order to build the knowledge for NLU. As an initial step, this work uses above mentioned sequence expansion labels to study the dendrograms and sequences of nodes longer than adjacency pairs.

The paper is structured as follows: Section 2 presents a summary of related literature and provides the necessary background. The Methodology and the clustering model is presented in Section 3, and Section 4 presents the results of our proposed model. Section 5 concludes this article.

2 Literature and Background

Structuring in dialogues have been explored by many researches utilising different sequencing theories: discourse representation theory (Kamp et al., 2011), conversation analysis (CA) (Sidnell and Stivers, 2012), and Rhetorical sequence theory (Hou et al., 2020) to name a few. Detailing these theories is beyond the scope of this work, but

Proceedings of the First Workshop on Computational Approaches to Discourse, pages 11–19
Online, November 20, 2020.
https://doi.org/10.18653/v1/P17

we will briefly explain some of their use-cases.

For instance, (Stent, 2000) used rhetorical sequence theory for sequencing task-driven dialogues and report several issues such as, deciding a minimal unit for annotation, overlap between subject-matter and presentational relations. In (Asher and Lascarides, 2003), the authors presented a novel theory called Segmented Discourse Interpretation Theory (SDRT), combining the knowledge from dynamic semantics, common sense reasoning, and speech act theory. The authors claimed SDRT to be the most formally mature and linguistically grounded theory.

While, the above mentioned works focused more on strengthening the theoretical foundations for dialogue sequencing, the authors (Boyer et al., 2009) identified themselves with solving practical matters of extracting sequences. Their corpus of human-human tutorial dialogues were manually annotated with dialogue acts and trained on a hidden Markov model (HMM) on adjacency pairs. More recently, the authors in (Nicholas et al., 2016) presented a multi-party corpus annotated with discourse sequence relations following SDRT mentioned earlier. Authors in (Gupta et al., 2018) proposed a hierarchical annotation scheme for query systems such as travel booking, in order to determine intents from complex nested queries compared to a single intent for each slot. In (Shi et al., 2019), the authors used a variational recurrent neural network (VRNN) and variational inference for dialogue sequence in task-oriented dialogues (finding restaurant and getting weather report).

The proposed work here is closely in line with (Zacharie et al., 2018; Duran and Battle, 2018; Tewari and Bensch, 2018), where in prior work the authors proposed a two step methodology of extracting two dimensional patterns in dialogues, followed by clustering. Their dialogues are manually annotated with emotion, gaze and dialogue act. In the latter work, the authors demonstrated the significance of dialogue sequencing for building domain agnostic dialogue models using CA. They explored sequence expansion and developed an annotation tool to annotate dialogues with sub-sequences based on CA and dialogue control functions. In the final work the authors used syntactic, communicative and CA based features and formalised them by extending the cooperating distributed grammar system.

The biggest difference of this work from the above mentioned prior works is in the definition of the task, i.e, the dialogue corpus. All the prior work has utilised either publicly available corpora based on query systems, while this work aimed to gather as diverse genres of task-driven query/reservation (booking laundry, ordering food), collaboration (cooking, taking medications, going to the flower shop) dialogues and chit-chat dialogues. The other difference is in the annotation approach and the training input, where, we neither use only manually labelled or the entire utterance as the input. Instead, we combine manually labelled and automatically extracted features.

The next section briefly provides some background on adjacency pairs and CA based sequence expansion.

2.1 Sequences in Dialogues

Adjacency pairs (Schegloff and Sacks, 1973) can be defined as utterances produced by two different participants and are adjacently placed. Instances of typically used adjacency pairs are greeting greeting, request accept/reject, offer accept/reject, question answer etc.

However, adjacency pairs allow *one-shot conversations* (McTear et al., 2016), where the human asks a question or queries a system and the system responds. Moving towards long and complex interactions which may include (pronoun resolution, topic management, etc) would leave adjacency pairs insufficient for the purpose. In the example below, we explain our scenario, labelled with dialogue control functions (Bunt, 1999), mentioned later.

Turn1 A1: Where is Eiffel Tower? Question

Turn2 Siri: Here is what I found. (displays information about Eiffel Tower) Answer

Turn3 A1: What are some of the good restaurants around it? Question

Turn4 Siri: Here is what I found. (displays restaurants around its current location) Incorrect Answer

Turn5 A1: Last year I had lot of fun in Scotland highlands. Can you tell me where is Windsor castle? Inform, Question

Turn6 Siri: I am sorry. Negative Feedback

This scenario poses at-least two challenges that motivates this work: *a)* at Turn3 'it' couldn't be be resolved by Siri and *b)* at Turn5 multiple dialogue control functions are present, where Siri fails to respond.

Research has been done already with regard to anaphora resolution using adjacency pairs (Palomar and Patricio, 2000), we propose to use sequence expansion for the problem *a)* above and for *b)* the annotation scheme proposed by Bunt et al. (Bunt et al., 2019). Next we explain the concept of sequence expansion (SE) to understand what do we mean by longer sequences.

Sequence Expansion (SE) (Stivers, 2012) constitutes labels that can precede, be inserted, or followed by the base adjacency pairs (introduced in Section 1). The above mentioned example can be translated with SE labels as in Table 1, and instead of knowledge from just a pair of turns, the machine can extract from multiple turns. Following such schemes allows machines, to have a longer window/slot for information. The other benefit is, it can optimise its knowledge and strategy, For example, if a machine observes that an SPP_{insert} is present in its slot, and its the machine's turn then it can switch the topic back to the base topic introduced at FPP_{base} if it hasn't been fulfilled by an SPP_{base}, etc.

To this end, SE labels are used to analyse the results of the clustering and to compare the amount of knowledge captured and the comprehensiveness they provide compared to adjacency pairs. The next section provides some details on the methodology employed by this work to predict distinctive clusters representing longer sequences.

3 Methodology

The method employed by this work to predict long sequences uses feature engineering and unsupervised clustering method on $n-$grams of syntactic features and dialogue control functions. The next sections provide details on the features used and the components of the model.

Overall, our framework consists of following stages represented in Figure 1:

1. Preparation of the corpus– consists of determining which genres should be considered, then merging of the samples from different sources was done, then the corpus was preprocessed by performing data cleaning, missing imputation, and assignment of unique-identifier.

2. Manual Annotation: transforming utterances to segments and labelling them with dialogue control function.

3. Extraction of features: next, a dependency parser was used on the corpus of dialogue segments to extract syntactic features ($uni-$grams and $tri-$grams).

4. Feature Transformation: employs a *tf-idf* when the feature consists of only dialogue control functions, and *GloVe* embeddings are used for different combinations of syntactic features and dialogue control functions.

5. Selection of features: we perform feature selection using PCA on the transformed features received from the previous stage.

6. Training of the model: the selected features are clustered with hierarchical agglomerative clustering.

7. Evaluation was done by computing Calinski Harabasz index, Silhouette score, Davies Bouldin score and Cophenetic Coefficient Correlation (Cophnet) for the clustering model.

3.1 Corpus

We conduct experiments on a collection of 78 dialogues of which 41 were synthetically created dialogues between an older adult H and a robot R. We used the scenario that R is situated in H's home to assist in daily tasks such as: meal reminders, playing board games, taking care of hazardous items etc.

The synthetic dialogues were combined with 9 dialogues from Dialog Bank [1] which already came with gold standard labels of ISO $24617-2$ scheme (Bunt et al., 2017) and 28 dialogues from dialogue breakdown detection challenge (DBDC3) (Higashinaka et al., 2017).

The synthetic dialogues and DBDC3 dialogues were hand labelled by the author with dialogue control functions following the ISO $24617-2$ annotation scheme. Since, this work is aimed towards extracting generic sequences hence, we combined different domains (taks-driven and chit-chat) and participant types (human-human, human-machine).

[1] https://dialogbank.uvt.nl/annotated dialogues/

Turn No./Participant	Utterances	DCF	SE
Turn1 A1:	Where is Eiffel Tower?	Question	$\mathbf{FPP}_{base}$
Turn2 Siri:	Here is what I found.	Answer	$\mathbf{SPP}_{base}$
Turn3 A1:	What are some of the good restaurants around it?	Question	$\mathbf{FPP}_{post}$
Turn4 Siri:	Here is what I found.	Incorrect Answer	$\mathbf{SPP}_{post}$
Turn5 A1:	Last year I had lot of fun in Scotland highlands.	Inform	$\mathbf{FPP}_{pre}$
Turn5 A1:	Can you tell me where is Windsor castle?	Question	$\mathbf{FPP}_{base}$
Turn6 Siri:	I am sorry.	Negative feedback	$\mathbf{FPP}_{insert}$

Table 1: The first column consists of the information about the turn and the participant, the second column provides one or more utterances with-in each turn, followed by the dialogue control functions (DCF) and sequence expansion (SE)

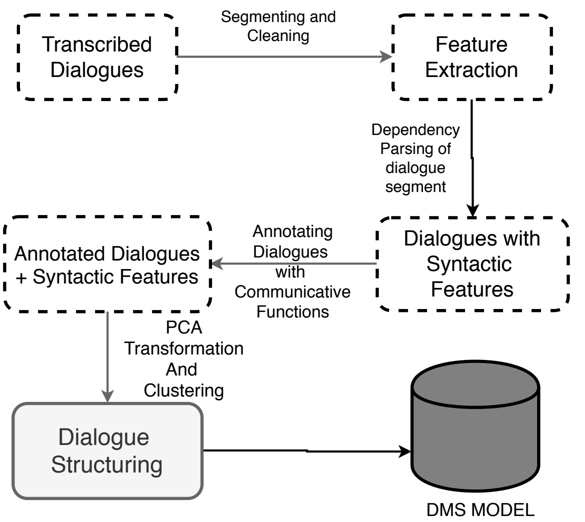

Figure 1: The workflow to obtain dialogue patterns for sequencing dialogues to build natural flows in DMS.

3.2 Syntactic Features and Dialogue Control Functions

We use dependency parsing for extracting syntactic features of types, $uni-$gram and $tri-$gram with dependency relationship. Dependency parsing generates syntactic sequences between **lexical** elements i.e(words), which are linked by binary *asymmetrical* relation called *dependencies*. Figure 2 illustrates a dependency parsing graph with syntactic sequence. This work uses Spacy dependency parser proposed in (Honnibal and Johnson, 2015).

Based on a manual analysis of dependency graphs on randomly selected samples from the corpus, we decided to use POS tags as $uni-$gram syntactic features: pronouns, proper nouns, direct object, indirect objects, coordinating conjunction, and interjection. For tri-gram syntactic features *(subject-object-verb)* tuples and dependency graphs

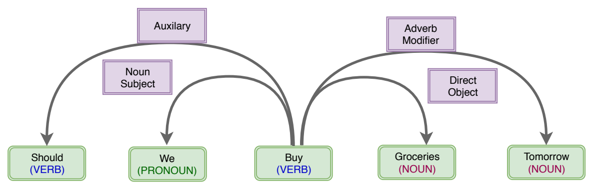

Figure 2: Dependency graph for an utterance, where coloured text in brackets are the POS tags associated to each lexical item (words). The arcs indicate the asymmetric dependency relation (auxiliary, noun subject, direct object and so on) between the head(arc orgins) and dependants(arc pointers).

of *(auxiliary verb)* and its right two neighbours were used.

Utterances in dialogues, have one to many relationship with functions to either provide or require information from an addressee and such functions are referred as dialogue control functions (Bunt et al., 2019). For instance in an utterance 'Hi John, Please get ready for some exercise' can be segmented into **'Hi John'** with dialogue control function (greeting) and **'Please get ready for some exercise'** with dialogue control function (request) and each of these segments are referred as *functional segments*. List of dialogue control functions used in this work are provided in Table 2.

3.3 Data Transformation and Reduction

Data transformation is an essential step for all machine learning algorithms and here we use two different transformation techniques for the two features used in this work.

Term Frequency Inverse Document Frequency tf-idf (Church and Gale, 1999) determines the relative frequency of terms in a document compared to the inverse proportion of that term over the collection of documents. Dialogue control functions are of categorical type and hence were trans-

Communicative Functions	Dialogue Control Functions
1.General Functions	Proposition, Set, Choice, Check Question, Inform, Agree, Disagree, Correction, Answer, Confirm, Dis-confirm, Promise, Offer, Address, Accept, Decline (Request, Suggest), Request, Instruct, Offer, Address, Accept, Decline (Offer).
2.Feedback Functs.	Auto-Positive, Allo-Positive, Auto-Negative, Allo-Negative, Feedback Elicitation.
3.Turn/Time Mgmt.	Accept-Turn, Grab-Turn, Assign-Turn, Keep-Turn, Release-Turn, Take-Turn, Stalling, Pausing.
5.Own/ Partner Comm. Mgmt.	Completion, Correct Misspeaking, Self-Error, Retraction, Self-correction.
6.Discourse Structuring	Interaction Structuring, Opening.
7.Social Obligation Mgmt.	Initial, Return (Greeting, Self-introduction, Goodbye), Apology, Thanking, Accept (Apology, Thanking).

Table 2: Different dialogue control functions corresponding to their respective Communicative functions

formed using tf-idf technique. Intuitively, it determines how significant a term is for a given document. Consider the corpus as a document collection D, with a term (dialogue control function) t, and document (a dialogue) $d \in D$, tf-idf can be calculated as (Ramos, 2003): $t_d = f_{t,d} \times \log(|D|/f_{t,D})$ Where, $f_{t,d}$ is the frequency of (dialogue control function) t in the given dialogue d, $|D|$ is the size of the corpus, and $f_{t,D}$ is the number of dialogues in which the dialogue control function t appears in the corpus D.

Word-embedding (Mikolov et al., 2013) transform words to vectors in a higher dimensional space to derive linear syntactic and/or semantic relationships between them. dc_t is the dialogue control function and $sf_t = [w_{1,t}, w_{2,t} ... w_{n,t}]$ are the $n-$gram syntactic features for each segment, where w is a single syntactic feature. The concatenation of these two features $F = [dc_t, sf_t]$ is the variable. dc_t and sf_t were averaged for each segment of an utterance resulting into $\bar{sf_t}$, $\bar{dc_t}$ and transformed using pre-trained GloVe (Pennington et al., 2014) embedding with 300 features providing $\bar{F} = [\bar{dc_t}, \bar{sf_t}]$, which is then given to PCA for feature selection, explained next.

Principal Component Analysis (PCA) reduces higher dimensional feature space to lower dimension, by selecting the features with highest variance (Shlens, 2014). PCA receives the above transformed features: tf-idf t_d and word-embedding $\bar{F}$. The linear transformations can be represented as a matrix computation: $P_1 t_d = T$ and $P_2 \bar{F} = F_{new}$. Where, the input to the HAC model are the rows of P_1 for only dialogue control functions and P_2 for combination of the features.

3.4 Hierarchical Agglomerative Clustering (HAC)

HAC is an unsupervised machine learning method (Murtagh and Contreras, 2012), that partitions the corpora into n singleton nodes and keeps merging mutually close pair of nodes until one final node is generated.

Let S_0 be the initial set of data points, at each step n_i is the new node formed by merging a_i and b_i with a given distance δ_i. It runs for $N1$ turns, resulting into a final state of only one node with all N initial nodes. Next we briefly describe the steps a HAC algorithm follows: *(i)* Generation of priority queue with nearest neighbours and minimal distances. *(ii)* Find the closest pair of nodes based on computed values for nearest neighbours and minimal distance, and append them to a list L to generate the dendrogram. *(iii)* Ensure the minimal distance between two nearest neighbours holds true till the end, and updates the minimum distance at every time step of the merging.

3.5 Model Definition

We experimented with four different HAC models and compared it for $Euclidean$ and $Manhattan$ distance measures for finding out the minimum distance between two feature combinations in order to merge them into clusters. To generate *dendrograms* we used $Ward$ linkage and $complete$ linkage. The four HAC models we experimented for different feature combinations: *(i)* Pre-defined number of clusters $n = 6$, distance metric: *Euclidean* distance, merging of clusters: *Ward*. *(ii)* Pre defined number of clusters $n = 5$, distance metric: *Euclidean* distance, merging of clusters: *Ward*. *(iii)* Pre defined number of clusters $n = 5$, distance metric: *Manhattan* distance, merging of clusters: *complete*. *(iv)* Pre defined number of clusters $n = 3$, distance metric: *Euclidean* distance, merging of clusters: *Ward*.

We ran the HAC models on tuple of features for each segment of an utterance. Following tuple of features were selected for running the experiment:

(i) only dialogue control functions *(DCF)*.

(ii) dialogue control functions and syntactic feature ($tri-$grams–subject-object-verb) as *(DCF,SS1)*.

(iii) dialogue control functions and syntactic feature ($tri-$grams–auxiliary verb, right neighbour1, right neighbour2) as *(DCF,SS2)*.

(iv) dialogue control functions and $uni-$gram syntactic features (Nouns, Direct object, Indirect object, Interjection and Coordinating Conjunction) *(DCF,ST)*.

(v) dialogue control functions and $tri-$gram syntactic features (auxiliary right neighbour1 Right neighbour2 and subject object verb) as (DCF,SS1,SS2).

(vi) dialogue control function and syntactic features, $uni-$grams and $tri-$grams as (DCF,ST,SS1,SS2).

4 Results

To evaluate the HAC model on different combination of features, we compute the silhouette coefficient, Calinski Harabasz index and Davies Bouldin score, these metrics illustrate if the model generated well defined clusters. In Statistics Cophnet, measures how well the *dendro-gram* preserves the pair wise distances of original data points (Saraçli et al., 2013). We use *Cophnet* to measure the correlation between original and the predicted data points.

The evaluation of the HAC model is illustrated in the Table 3. The overall performance of the HAC model is good on specific combination of features **DCF, SS1, SS2** and **DCF, ST** as highlighted in bold with high Calinski Harabasz Index, Silhouette score, and Davies Bouldin score. The Cophnet score is high for half of the combination of features i.e, **DCF, SS1**, **DCF, ST**, and **DCF, SS1, SS2**. We can see that the performance of the HAC model on only **DCF** is also high, however it is not a relevant result for us because it doesn't convey any information about the sequence.

4.1 Empirical Analysis of HAC Model

In order to identify the sequence expansions we manually analysed random sample of dendrograms, for all the six combination of features mentioned above with 200 nodes. We provide here five such examples of the analysed dendrograms, which are manually labelled with sequence expansion labels, in-order to see if such labelling can help to capture and build more knowledge.

Table 4 provides two examples extracted from one of the generated dendrograms, for (dialogue control functions and $uni-$gram syntactic feature). In the first sample, *Instruct* node with the syntactic feature *mill* was adjacent to *Question* node with the syntactic feature *picket*, other adjacent nodes without any syntactic feature was a positive feedback and an answer. Indicating that this example could possibly be a part of a navigation instruction, while the other dialogue seems to be a part of a chit-chat dialogue. For each example, each subsequent line represents the closest node while browsing the dendrogram from top to bottom if its vertically drawn. As it can be seen in these examples, the model doesn't predict the nodes to be in perfect pairs, hence highlighting that using adjacency pairs will be insufficient in extracting knowledge that is not distributed with-in pairs.

Three examples from the analysed dendrograms are presented in Table 5 representing the combination of features (dialogue control function and $tri-$gram syntactic feature). Also, for this case we manually labelled them with SE labels. The example number 1 seems to be about finding glasses, while the others indicate towards them being a part of a dialogue on machines and stealing of the jobs.

Sr.No	Feature Combination	Calinski Harabasz Index	Silhouette score	Cophnet score	Davies Bouldin
1.	DCF	*81333*	*0.77*	*0.40*	*0.35*
2.	DCF SS1	*3910*	*0.51*	***0.76***	*0.59*
3.	DCF SS2	***16454***	*0.60*	*0.56*	*0.54*
4.	DCF ST	***11002***	***0.73***	***0.80***	***0.20***
5.	DCF SS1 SS2	***22096***	***0.66***	***0.75***	*0.50*
6.	DCF ST SS1 SS2	*3550*	*0.60*	*0.64*	*0.56*

Table 3: Evaluation of HAC Model on eight combination of features with communication and syntax features.

S.No	Feature Combination	Sequence Expansion
1.	Positive feedback, uh huh	FPP_{pre}
	Instruct, mill	SPP_{pre}
	Check question, picket, fence	FPP_{base}
	Positive feedback, answer, picket	SPP_{base}
	Positive feedback, uh huh	FPP_{post}
2.	Inform, school	FPP_{pre}
	Question, kids	FPP_{base}
	Stalling, uh	FPP_{insert}

Table 4: Some clusters from HAC model for the combination of features *dialogue control functions and syntactic features*

Here, it can be found in Example 2 third line that there is no FPP_{base} for the SPP_{base}, indicating that the parts for the same pair (base, pre, post, insert) can sometimes be very far away or possibly the model places them far because of the dissimilarities between them.

The analysis also showed that among the syntactic features, $uni-$grams were present dominantly around 84% of the times, while $tri-$grams of subject-object-verb tuples constituted 50% of the segments and auxiliary verbs were 20% of the segments.

5 Conclusion, Discussion and Future Work

This work explored combination of features (syntactic features and dialogue control functions) in order to find sequences in dialogues, such that we can build NLU functions for capturing information distributed over turns longer than two for DMs to possibly conduct flexible dialogues. Dependency parsing was used for extracting syntactic features ($uni-$grams and $tri-$grams) and dialogue control functions were labelled manually using ISO $24617-2$ scheme. The feature transformation was done using tf-idf (when using only dialogue control function training the model), and GloVe embedding were used for combination of features (dialogue control functions and syntactic features), for both the cases feature selection was done with PCA. The selected features were modelled with hierarchical agglomerative clustering, the results validated our assumption that capturing longer sequences using syntactic features can provide knowledge that adjacency pairs would fall short in.

This work being at a preliminary stage doesn't provide any concrete solution yet for building flexible dialogue strategies and rich knowledge sources, however it can be seen as more of a proof-of-concept for using syntactic features and sequence expansion labels for dialogue sequencing. The benefit of using syntactic features is that they can be extracted automatically from the raw data and state-of-the-art methods are robust enough. This work explored tuples of syntactic features, instead trees or graphs must be explored. Syntactic features provides flexibility to a machine, in the sense that it can select and prioritise to accomplish a topic (objects, nouns, etc) depending on the goals and/or the domain it is employed for. For pronoun resolution, relationship between prior mentioned proper noun/s and incoming pronouns can be established

S.No	Feature Combination	Sequence Expansion
1.	Inform, cant see anything	FPP_{pre}
	Question, do you remember	FPP_{base}
	Inform, on the bedside	FPP_{insert}
	Inform, did't find glasses	SPP_{insert}
2.	Turn keep, don't you see	FPP_{insert}
	Confirm, they do not	SPP_{insert}
	Accept, machines steal jobs	SPP_{base}
	Inform, a set people	FPP_{pre}
3.	Retract, it does not, steals jobs	SPP_{base}
	Inform, machines	FPP_{pre}
	Question, work that does	FPP_{base}

Table 5: Selection of clusters from HAC model indicating sequence expansions for feature combination *dialogue control function and* $tri-gram$ *syntactic features.*

using extraction of uni-gram syntactic features delimited by SE labels. For managing multiple dialogue control functions, coordinating conjunctions and interjections can be used for identifying response generation.

This work also comes with its limitations, where the first is related to the corpus, which could be biased due to a large number of samples being synthetically prepared by the author. Another limitation is the size of the corpus. The author is currently working on both of these limitations and in the future we have planned to combine different genres of dialogues from publicly available sources. Another limitation of this work is that it doesn't use any dialogue features such as intents, semantics, context, etc. Other limitations include selection and model of the syntactic features, where some of the features such as auxiliary verbs should be dropped because of their low frequency, it could be also a bias from the corpus that was used. A common assumption that dialogues are about subjects objects and verbs could not be held by this work.

Whether dialogues are task-driven or open ended or chit-chat– one commonality is that they all are directed towards activities fulfilling human needs (both tangible or intangible) More abstract models such as BDI models (Rao et al., 1995) and/or Activity theory (Leontiev, 1978) should be considered and be complemented with syntactic and pragmatic features mentioned here.

6 Acknowledgement

I would like to thank Associate Prof. Suna Bensch at the Department of Computing Science, Umeå University, Umeå, Sweden for her intellectual contribution towards ideation and refining of the research work done in this article.

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 721619 for the SOCRATES project.

References

Nicholas-Michael Asher and Alex Lascarides. 2003. *Motivating Rhetorical Relations*, chapter 1. Cambridge University Press.

Kristy Boyer, Robert Phillips, Young Ha Eun, Michael Wallis, Mladen Vouk, and James Lester. 2009. Modeling dialogue structure with adjacency pair analysis and hidden markov models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 49–52. ACL.

Harry Bunt. 1999. Dynamic interpretation and dialogue theory. *The structure of multimodal dialogue*, 2:1–8.

Harry Bunt, Volha Petukhova, Andrei Malchanau, Alex Chengyu Fang, and Kars Wijnhoven. 2019. The dialogbank: dialogues with interoperable annotations. *Language Resources and Evaluation*, 53:213–249.

Harry Bunt, Volha Petukhova, David Traum, and Jan Alexandersson. 2017. *Dialogue Act Annotation with the ISO 24617-2 Standard*, pages 109–135. Springer International Publishing.

Kenneth Church and William Gale. 1999. Inverse document frequency (idf): A measure of deviations from poisson. In *Natural language processing using very large corpora*, pages 283–295. Springer.

Nathan Duran and Steve Battle. 2018. Conversation analysis structured dialogue for multi-domain dialogue management. *DEXAHAI*, pages 1–4.

Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 6, Belgium. ACL.

Ryuichiro Higashinaka, Funakoshi Kotaro, Inab Michimasa, Tsunomori Yuiko, Takahashi Tetsuro, and Kaji Nobuhiro. 2017. Overview of dialogue breakdown detection challenge 3. *Proceedings of Dialogue System Technology Challenge*, page 14.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Shengluan Hou, Shuhan Zhang, and Chaoqun Fei. 2020. Rhetorical structure theory: A comprehensive review of theory, parsing methods and applications. *Expert Systems with Applications*, 157:113421.

Hans Kamp, Josef Van Genabith, and Uwe Reyle. 2011. *Discourse Representation Theory*, pages 125–394. Springer Netherlands, Dordrecht.

Aleksei Nikolaevich Leontiev. 1978. *Activity, consciousness, and personality.* Prentice-Hall, Moscow, Russia.

Michael McTear, Zoriada Callejas, and Davis Griol. 2016. Towards a technology of conversation. In *The Conversational Interface*, chapter 3, pages 25–45. Springer.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA. Curran Associates Inc.

Fionn Murtagh and Pedro Contreras. 2012. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97.

Asher Nicholas, Hunter Julie, Morey Mathieu, Benamara Farah, and Afantenos Stergos. 2016. Discourse Structure and Dialogue Acts in Multiparty Dialogue: the STAC Corpus. In *10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2721–2727, Portoroz, Slovenia.

Manuel Palomar and Martínez-Barco Patricio. 2000. Anaphora resolution through dialogue adjacency pairs and topics. In *Natural Language Processing — NLP 2000*, pages 196–203, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. ACL.

Juan Ramos. 2003. Using tf-idf to determine word relevance in document queries. volume 242, pages 133–142.

Anand S Rao, Michael P Georgeff, et al. 1995. Bdi agents: from theory to practice. In *ICMAS*, volume 95, pages 312–319, USA. MIT Press.

Sinan Saraçli, Nurhan Doğan, and İsmet Doğan. 2013. Comparison of hierarchical cluster analysis methods by cophenetic correlation. In *Journal of Inequalities and Applications*, 1, page 203. SpringerOpen.

Emanuel A Schegloff and Harvey Sacks. 1973. Opening up closings. In *Semiotica*, volume 8, pages 289–327. Walter de Gruyter.

Weiyan Shi, Tiancheng Zhao, and Zhou Yu. 2019. Unsupervised dialog structure learning.

Jonathon Shlens. 2014. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, page 12.

Jack Sidnell and Tanya Stivers. 2012. *The Handbook of Conversation Analysis.* Wiley-Blackwell, UK.

Amanda Stent. 2000. Rhetorical structure in dialog. In *In Proceedings of the 2nd International Natural Language Generation Conference (INLG'2000*, pages 247–252.

Tanya Stivers. 2012. Sequence organization. In *The Handbook of Conversation Analysis*, chapter 10, pages 191–209. Wiley-Blackwell, UK.

Maitreyee Tewari and Suna Bensch. 2018. Natural language communication with social robots for assisted living. In *IROS Workshop in Robots for Assisted Living*, pages 1–4, Madrid, Spain.

Ales Zacharie, Pauchet Alexandre, and Knippel Arnaud. 2018. Extraction and clustering of two-dimensional dialogue patterns. *International Journal on Artificial Intelligence Tools*, 27(02):1850001.